International Journal of Future Generation Communication and Networking
Vol. 15, No. 1, (2022), pp. 90–99

Enhanced Neural Network Based Combined Classification for Intrusion Detection System

Nusrath Unnisa A[1], Dr.Manjula Yerva[2], Dr. M. Z. Kurian[3]

[1]PG Student, Dept. of ECE., Sri Siddhartha Institute of Technology, Tumakuru, Karnataka, India, nussukhanum@gmail.com
[2]Assistant Professor, Dept. of ECE., Sri Siddhartha Institute of Technology, Tumakuru Karnataka, India ,manjulayerva@ssit.edu.in
[3]Head of the Department, Dept. of ECE., Sri Siddhartha Institute of Technology, Tumakuru Karnataka, India, mzkurianvc@yahoo.com

Abstract

Communication is a method of conveying information between end users. In this digital era data communication plays a vital role where the information is transmitted through internet. All over the world, web applications are used to store sensitive and personal data. These issues make them an attractive target for malware that exploits vulnerabilities in order to acquire unauthorized access. In order to access a particular resource a unique address is used called URL (Uniform Resource locator). These URL's can be attacked by an intruder to access the information stored on the client side or server by redirecting them. This paper proposes a model for intrusion detection system which is a combination of K-Nearest neighbour, Support vector machine and neural network in which the final output class is decided as Malicious or Benign based on the majority voting scheme. The work is carried out by using Cross Site scripting (XSS) attacks database and an alert is given to the end user that URL is malicious or benign.

Keywords: Artificial Neural Network, Cross Site Scripting, Intrusion detection System, K-nearest Neighbour, Support vector machine, Uniform resource locator.

1. Introduction

Most initial computer applications had very little security, until data was considered to be useful, but not something to be protected. In order to handle financial and personal data the real need for security was felt like never before. The main aim of network security is to provide integrity, authentication, non-repudiation and authentication.

No matter how much secure a system is made there would be attackers who would constantly try to find their way and we call them as intruders because they try to intrude into the privacy of the network. In order to alert about the intrusion an intrusion detection system [1] is used which acts as a heart of the network. IDS can be classified into two categories such as statistical anomaly detection and rule based detection.

ISSN: 2233-7857 IJFGCN

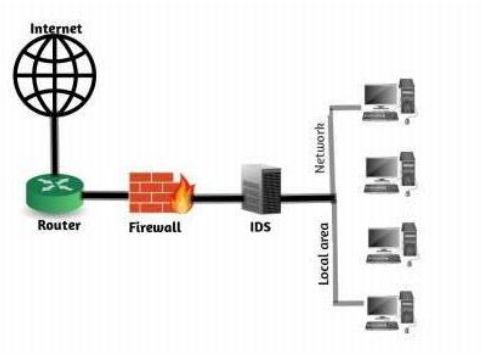


Figure 1: An intrusion detection System model

Statistical anomaly detection: In this case, behaviour of users over time is captured as statistical data and processed. Rules are applied to test whether the user behaviour is legitimate or not. This can be done in two ways.

- Threshold detection: In this type, thresholds are defined for all the users as a group and frequency of various events is measured against these thresholds.
- Profile based detection: In this type, profiles for individual users are created and they are matched against the collected statistics to see if any irregular patterns emerge.

Rule based detection: A set of rule is applied to see if a given behaviour is suspicious enough to be classified as an attempt to intrude. This is also classified into two subtypes.

- Anomaly detection: Usage patterns are collected to analyse deviation from these usage patterns with the help of certain rules.
- Penetration identification: This is an expert system that looks for illegitimate behaviour.

All over the world, web applications are used to store sensitive and personal data. These issues make them an attractive target for malware that exploits vulnerabilities in order to acquire unauthorized access. They are therefore easily exploited by malware to obtain unauthorized information. SQL injections and cross-site scripting are examples of such attacks. The victim can be affected by XSS by stealing cookies, Web page modification, clipboard contents-capture, key logging, port scanning, etc. The safety of web applications is therefore paramount. The lack of verification of client input or the environment is the most common security vulnerability of web applications, which are repeatedly discovered and exploited on both the client and server sides. There are three methods for detecting and protecting against XSS attacks, such as

- Statistical analysis: An analysis of this type may provide formal guarantees that certain vulnerabilities will not occur, but it may also be slow or unproductive.
- Dynamic analysis: The goal of this analysis is to understand how the script behaves during execution. This analysis involves switching the interpreter or checking the syntactic structure;
- Machine learning: The use of machine learning is to build classifiers and predict based on the knowledge of the scripts available.

The remaining section of the paper is organised such as section 2 deals with related work, in Section 3 proposed method is discussed, section 4 contains details on Cross site scripting dataset , section 5 is about implementation method , section 6 deals with experiments and results and conclusion is written in section 7.

ISSN: 2233-7857 IJFGCN

2. Related Work

Research on intrusion detection system is never ending. Scholars are working to find the best and efficient method of finding attacks and classifying them as malicious or benign. Alsharafat, W [3] has proposed a method in which artificial neural network and extended classifier system is used for network intrusion detection. Another method was proposed by Hajimirzaei, B.; Navimipour[4] which works to find intrusion detection for cloud computing using neural networks.

Based on statistical pre-processing and neural network a hierarchical network intrusion detection system was developed which is proposed by Zhang, Z., Li, J.[5] .S. Kumar, A. Yadav [6] has done research on increasing performance of intrusion detection system using neural network. Based on hybrid distance sum based support vector machine technique an IDS system was proposed by Guo, c., Zhou, Y[7].

By using tree based classifier, Thaseen, S., & Kumar, C.[8] proposed a method by selecting random patterns. Another method was proposed by J. V. Anand Sukumar et al [9], which uses improved Genetic K means algorithm to detect intrusion. By considering Genetic Algorithm, an intrusion detection system was proposed by MS Hoque, et al [10]. By using KNN and SVM classifier on cross site scripting dataset a method was proposed by Fawaz mereani [11].

3. Proposed Method

In the proposed method , classifiers such as KNN and SVM are used along with the neural network.

3.1 K-Nearest Neighbour classifier

The sample is said to have probability of belonging to the class if most of its neighbours belong to the same class. The performance of KNN is dependent on parameter K which is found by the user.

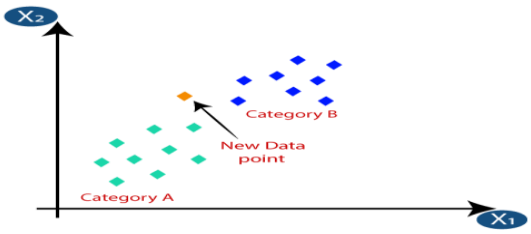


Figure 2: KNN classifier

3.2 Support Vector Machine Classifier

The procedure of using SVM is to differentiate max margin hyperplane in the n-dimension feature space.

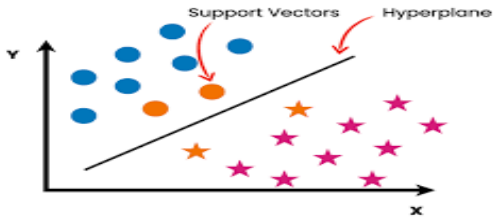


Figure 3: SVM classifier

ISSN: 2233-7857 IJFGCN

3.3 Neural Network

Neural network reflects the behaviour of human brain, which is derived by imitating the way that biological neurons signal to one another [2]. A basic block diagram for artificial neural network is shown in figure 4.

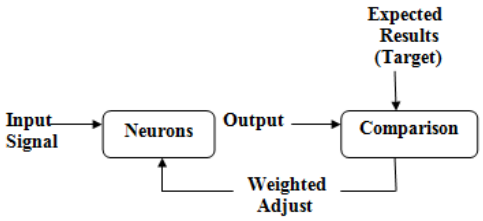


Figure 4: Block Diagram –ANN for Pattern classification

Artificial neural network consists of one input layer, one or more hidden layers and one output layer. Each node connected to another and it has a connected weight and threshold. When the output of an individual node is crossing the threshold value which is set, that node is activated and sends data to the next layer of the network, otherwise no data is passed along to the next layer of the network. Figure 5 shows a basic ANN model.

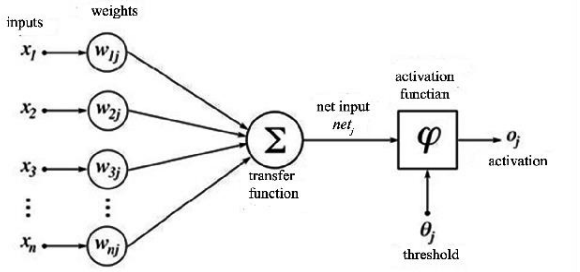


Figure 5: ANN Model

An ANN model consists of input data, weights, bias and an output. The formula can be written as

$\sum W_i X_i$ + bias = $W_1 X_1 + W_2 X_2 + W_3 X_3$ + bias

Output = f(x) = $\begin{cases} 1 \text{ if } \sum W_1 X_1 + b >= 0; \\ 0 \quad \text{if } \sum W_1 X_1 + b < 0; \end{cases}$

4. Cross site scripting (XSS) Datasets

Cross site scripting (XSS) is one of the injection based attacks found in web applications. A hacker can use XSS to send a malicious script to an unsuspecting user. An URL (Uniform Resource Locator) is a unique identifier used to locate a resource on the Internet. URL protocols include HTTP (Hypertext Transfer Protocol) for web resources. The browser used by the end user has no idea that the script is malicious and he allows the script to be executed. The malicious script may contain cookies, session tokens or any other information which is sensitive, retained by the browser and used with that site.

ISSN: 2233-7857 IJFGCN

The datasets used are available online and downloaded from the developer site [12-15]. Totally 4000 URL information is available which are trained to get the classification such as malicious or benign. The dataset taken are divided to use for training and testing respectively. The features used can be classified as structural and behavioural.

- Structural features include alphanumeric characters present in the script whereas behavioural features include command and functions in the script. Table 1 show the different features used.

Table 1: Description of Features

Sl No	Features	Features type	Description
1	Symbols	Structural	`,!,@,#,$,%,^,&,*,<,>,?,/,[,{,],}:,;,”,”,’,’,\,|,-,~,+,_,(,)
2	Combination of Punctuations	Structural	><’, “><,[],==,&#
3	Objects	Behavioural	Document, window, iframe, location, This
4	Tags	Behavioural al	DIV, IMG, <script

5. Implementation Method

In the proposed method, URL is taken as input whose features are extracted and fed to the KNN, SVM classifier and for neural network and also fed to the proposed classifier.

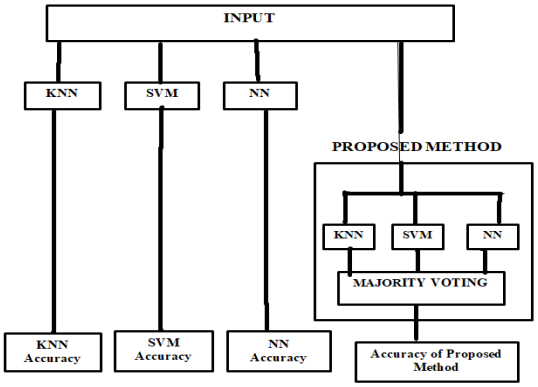


Figure 6: Block Diagram for implementation

Based on the training data the individual classifier will give the output as malicious or benign, whereas the proposed method gives the output by taking majority voting. Figure 6 shows the block diagram used for implementation. The implementation flow diagram is shown in Figure 7 below.

ISSN: 2233-7857 IJFGCN

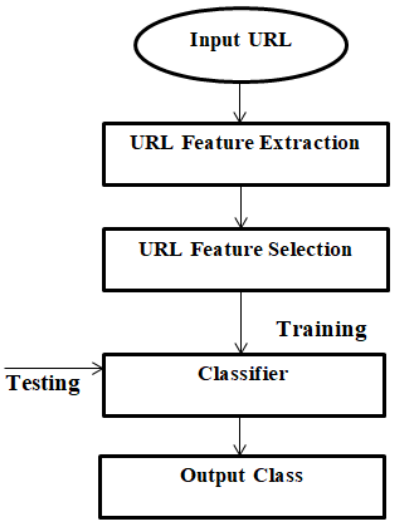


Figure 7: Flow Diagram for implementation

6. Results and Discussion

Matlab R2017a is used to conduct the experiments on XSS dataset. Datasets are present in excel sheets which contains 4000 URL information. These URL's are trained using ANN and out of 4000 only 300 URL are taken for testing purpose and the results are obtained such as classification of the URL as malicious or benign.

A confusion matrix is generated for the selected URL and the different parameters are calculated such as accuracy, Prevalence, TPR, Precision, F1 score, and FRP. To describe the performance of a classification mode or classifier, a confusion matrix is written on a set of test data for which the true value are known as shown in figure 8.

		Prediction	
		Positive	**Negative**
Actual	**Positive**	True Positive(TP)	False Negative (FN)
	Negative	False Positive(FP)	True Negative(TN)

Figure 8: Confusion matrix

Parameters can be calculated as shown below:

Precision: To decide, how often is it correct when it predicts yes.

$$\text{Precision} = \frac{TP}{Predicted\ Yes} \text{---------- (1)}$$

Prevalence: to decide how often does the yes condition actually occur in our sample. It can be calculated as

$$\text{Prevalence} = \frac{Actual\ Yes}{Total} \text{---------- (2)}$$

ISSN: 2233-7857 IJFGCN

Accuracy: To decide how often is the classifier correct in overall. It is given by the formula

Accuracy = $\frac{TP+TN}{Total}$---------- (3)

True Negative Rate: To decide how often it predict no, when its actually no. it can be calculated as,

True Negative Rate= $\frac{TN}{Actual\ No}$------ (4)

False positive Rate: To decide, how often it predicts yes when it is actually no. It is calculated as

False positive Rate = $\frac{FP}{Actual\ No}$------- (5)

True positive Rate: To decide how often it predict Yes, when it is actually yes. It is calculated as

True positive Rate = $\frac{TP}{Actual\ Yes}$------ (6)

To determine the harmonic mean of precision and recall, F1 score is used.

Accuracy calculation for KNN classifier, SVM classifier, and neural network and for the proposed method is shown in figure 9. KNN is represented by 1, 2 for SVM, 3 for NN and 4 for the proposed method.

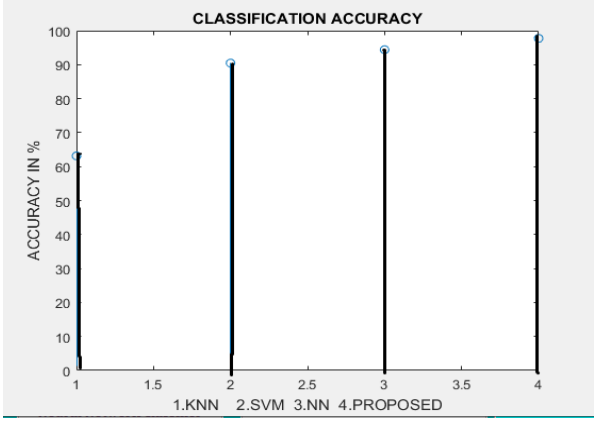


Figure 9: Accuracy for different classifiers

The pattern recognition used in neural network is shown in figure 10.

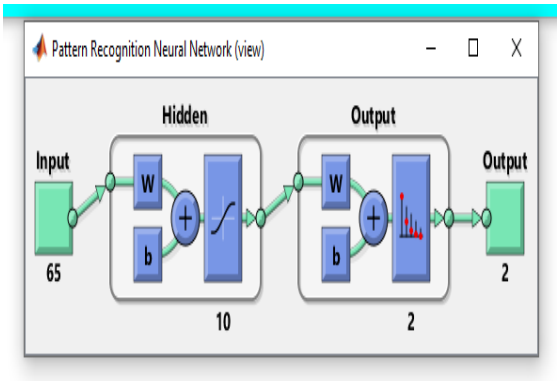


Figure 10: Pattern recognition in NN

ISSN: 2233-7857 IJFGCN

Figure 11 shows the output window in which the status of the URL is shown such as Malicious or Benign and figure 12 gives the Confusion Matrix.

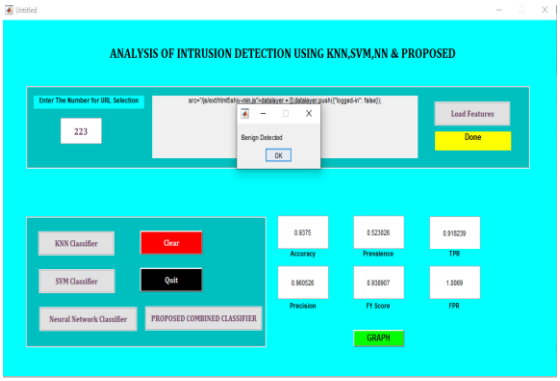


Figure 11: Output window

Figure 12: Confusion matrix

The performance analysis such as accuracy and precision for KNN, SVM, NN and for the proposed method is shown in Table 2.

Table 2: Performance Analysis

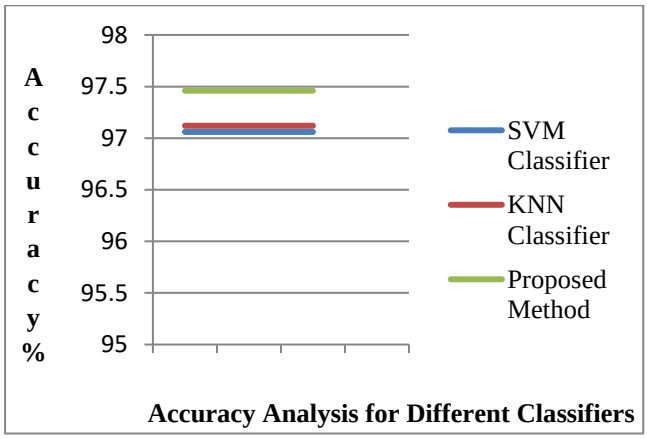
	Accuracy (%)	**Precision (%)**
KNN	63.15	77.77
SVM	90.46	84.74
NN	94.40	96.05
Proposed Method	97.46	97.74

ISSN: 2233-7857 IJFGCN

6.1 Discussion

The work carried out in this paper is compared with reference paper [11] and showed the results in Table 3 with corresponding charts shown in figure 13.

Table 3: Performance Comparison



Reference Paper	**Accuracy (%)**
SVM (Polynomial Kernel) [11]	97.06%
k-NN classifier [11]	97.12%
Proposed Method	**97.46%**

Figure 13: Accuracy Comparison

Conclusion

With the improvement in technology, people are relying more on internet and that is where hackers are injecting their malicious script to retrieve the stored information on browser. This paper demonstrates an intrusion detection system using a hybrid model to detect such attacks and alerting the user that the network traffic is malicious or benign. To carry out the work Cross site scripting attacks dataset are used which are popularly used Web attacks. Using the proposed Hybrid model gives the better results compared to other classifiers such as KNN and SVM.

References

1. A. B. Mohamed, N. B. Idris, and B. Shanmugum, “A brief introduction to intrusion detection system,” in Communications in Computer and Information Science, 2012, vol. 330 CCIS, pp. 263–271
2. L. P. Dias, J. J. F. Cerqueira, K. D. R. Assis, and R. C. Almeida, “Using artificial neural network in intrusion detection systems to computer networks,” 2017 9th Comput. Sci. Electron. Eng. Conf. CEEC 2017 -Proc., pp. 145–150, 2017.
3. Alsharafat, W. (2013): Applying Artificial Neural Network and eXtended Classifier System for Network Intrusion Detection. International Arab Journal of Information Technology 10(3)
4. Hajimirzaei, B.; Navimipour, N. J. (2019): Intrusion detection for cloud computing using neural networks and artificial bee colony optimization algorithm. ICT Express 5(1), pp. 56-59.
5. Zhang, Z., Li, J., Manikopoulos, C. N., Jorgenson, J., &Udes, J. (2001, June). HIDE: A Hierarchical Network Intrusion Detection System Using Statistical Preprocessing And Neural Network Classification. In Proc. IEEE Workshop on Information Assurance and Security, pp. 85-90.
6. S. Kumar, A. Yadav. Increasing Performance of Intrusion Detection System Using Neural Network. 2014 IEEE International Conference on Advanced Communication Control and Technologies (ICACCCT), pp. 1935-1939.

ISSN: 2233-7857 IJFGCN

7. Guo, c., Zhou, Y., Ping, Y., Zhang, Z., Liu, G., &Yang, Y. (2014). A Distance Sum-Based Hybrid Method for Intrusion Detection.Applied Intelligence, 40(1), 178-188.
8. Thaseen, S., & Kumar, C. (2013, February). An Analysis of Supervised Tree Based Classifiers for Intrusion Detection System. IEEE International Conference on In Pattern Recognition, Informatics and Medical Engineering (PRIME), 2013, pp. 294-299.
9. J. V. Anand Sukumar, I. Pranav, M. M. Neetish, and J. Narayanan, "Network Intrusion Detection Using Improved Genetic k-means Algorithm," 2018 Int. Conf. Adv. Comput. Commun. Informatics, ICACCI 2018, pp. 2441–2446, 2018.
10. MS Hoque, M Mukit, M Bikas, A Naser, An Implementation of Intrusion Detection System using Genetic Algorithm, *International Journal of Network Security and its Applications*, **2012**, 4(2), 109-120.
11. Fawaz Mereani, Jacob Howe, "Detecting cross site scripting attacks using Machine learning", DOI: 10.1007/978-3-319-74690-6_20.
12. Examples of malicious javascript (2014). https://aw-snap.info/articles/js-examples.php. Accessed 16 Dec 2016.
13. Karnad, K.: XSS payloads you may need as a pen-tester (2014). https://www.linkedin.com/pulse/20140812222156-79939846-xss- vectors-you-may-need-as-a-pen-tester. Accessed 25 Dec 2016.
14. XSS Payloads: XSS payloads you may need as a pen-tester. http://www.xsspayloads.com/payloads.html.Accessed 14 Oct 2016.
15. Fernandez, K., Pagkalos, D.: XSS (Cross-Site Scripting) information and vulnerablewebsites archive. XSSed.com. Accessed 14 June 2017.

ISSN: 2233-7857 IJFGCN